EXHIBIT O

Transport layer

From Wikipedia, the free encyclopedia

In computing and telecommunications, the **transport layer** is the second highest layer in the four and five layer TCP/IP reference models, where it responds to service requests from the application layer and issues service requests to the Internet layer. It is also the name of layer four of the seven layer OSI model, where it responds to service requests from the session layer and issues service requests to the network layer. The definitions of the transport layer are slightly different in these two models. This article primarily refers to the TCP/IP model. See also the OSI model definition of the transport layer.

A **transport protocol** is a protocol
on the transport layer. The two most widely used transport protocols on the Internet are the connection oriented TCP (Transmission Control Protocol), and UDP (User Datagram Protocol). TCP is the more complicated and most common. Other options are the Datagram Congestion Control Protocol (DCCP) and Stream Control Transmission Protocol (SCTP).

OSI Model

7 Application layer
6 Presentation layer
5 Session layer
4 **Transport layer**
3 Network layer
2 Data link layer

- LLC sublayer
- MAC sublayer

1 Physical layer

The five-layer TCP/IP model

5. Application layer

DHCP * DNS * FTP * Gopher * HTTP * IMAP4 * IRC * NNTP * XMPP * POP3 * SIP * SMTP * SNMP * SSH * TELNET * RPC * RTP * RTCP * RTSP * TLS/SSL * SDP * SOAP * BGP * PPTP * L2TP * GTP * STUN * NTP * ...

4. Transport layer

TCP * UDP * DCCP * SCTP * RSVP * ...

3. Internet Layer

IP (IPv4 * IPv6) * IGMP * ICMP * OSPF * ISIS * IPsec * ARP * RARP * RIP * ...

2. Data link layer

802.11 * ATM * DTM * Token Ring * Ethernet * FDDI * Frame Relay * GPRS * EVDO * HSPA * HDLC * PPP * ...

1. Physical layer

Ethernet physical layer * ISDN * Modems * PLC * SONET/SDH * G.709 * Optical Fiber * WiFi * WiMAX * Coaxial Cable * Twisted Pair * ...

The transport layer is typically handled by processes in the host computer operational system, and not by routers and switches. The transport layer usually turns the unreliable and very basic service provided by the Network layer into a more powerful one.

In the TCP/IP model, the transport layer is responsible for delivering data to the appropriate application process on the host computers. This involves statistical multiplexing
of data from different application processes, i.e. forming data packets, and adding source and destination port numbers in the header of each transport layer data packet. Together with the source and destination IP address, the port numbers constitutes a network socket, i.e. an identificiation address of the process-to-process communication. In the OSI model, this function is supported by the session layer.

Some transport layer protocols, for example TCP but not UDP, support virtual circuits, i.e. provide connection oriented communication over an underlying packet oriented datagram
network. A byte-stream is delivered while hiding the packet mode communication for the application processes. This involves connection establishment, dividing of the data stream into packets called segments, segment numbering and reordering of out-of order data.

Finally, some transport layer protocols, for example TCP but not UDP, provides end-to-end reliable communication, i.e. error recovery by means of error detecting code and automatic repeat request (ARQ) protocol. The ARQ protocol also provides flow control, which may be combined with congestion avoidance.

UDP is a very simple service, and does not provide virtual circuits, nor reliable communication, leaving these to the application. The UDP packets are called datagrams rather than segments.

TCP is used for for example HTTP web browsing and email transfer. UDP may be used for multicasting and broadcasting, since retransmissions are not possible to a large amount of hosts. UDP typically gives higher throughput and shorter latency, and is therefor often used for realtime multimedia communication where packet loss occasionally can be accepted, for example IP-TV and IP-telephony, and for online computer games.

In many non-IP-based networks, for example X.25, Frame Relay and ATM, the connection oriented communication is implemented at network layer or data link layer rather than the transport layer. In X.25, in telephone network modems and in wireless communication systems, reliable node-to-node communication is implemented at lower protocol layers.

In the OSI/X.25 protocol suite, there are five classes of the OSI transport protocol, ranging from class 0 (which is also known as **TP0** and provides the least error recovery) to class 4 (which is also known as **TP4** and is designed for less reliable networks, similar to the Internet).

List of transport layer services

There is a long list of services that can be optionally provided by the transport layer. None of them are compulsory, because not all applications want all the services available. Some can be wasted overhead, or even counterproductive in some cases.

Connection-oriented
: This is normally easier to deal with than connection-less models, so where the Network layer only provides a connection-less service, often a connection-oriented service is built on top of that in the Transport layer.

Same Order Delivery
: The Network layer doesn't generally guarantee that packets of data will arrive in the same order that they were sent, but often this is a desirable feature, so the Transport layer provides it. The simplest way of doing this is to give each packet a number, and allow the receiver to reorder the packets.

Reliable Data
: Packets may be lost in routers, switches, bridges and hosts due to network congestion, when the packet queues are filled and the network nodes have to delete packets. Packets may be lost or corrupted in for example Ethernet due to interference and noise, since Ethernet does not retransmit corrupt packets. Packets may be delivered in the wrong order by an underlying network. Some transport layer protocols, for example TCP, can fix this. By means of an error detection code, for example a checksum, the transport protocol may check that the data is not corrupted, and verify that by sending an ACK message to the sender. Automatic repeat request schemes may be used to retransmit lost or corrupted data. By introducing segment numbering
: in the transport layer packet headers, the packets can be sorted in order. Of course, error free is impossible, but it is possible to substantially reduce the numbers of undetected errors.

Flow Control
: The amount of memory on a computer is limited, and without flow control a larger computer might flood a computer with so much information that it can't hold it all before dealing with it. Nowadays, this is not a big issue, as memory is cheap while bandwidth is comparatively expensive, but in earlier times it was more important. Flow control allows the receiver to say "Whoa!" before it is overwhelmed. Sometimes this is already provided by the network, but where it is not, the Transport layer may add it on.

Congestion avoidance
: Network congestion
: occurs when a queue buffer of a network node is full and starts to drop packets. Automatic repeat request may keep the network in a congested state. This situation can be avoided by adding congestion avoidance to the flow control, including slow-start. This keeps the bandwidth consumption at a low level in the beginning of the transmission, or after packet retransmission.

Byte orientation
: Rather than dealing with things on a packet-by-packet basis, the Transport layer may add the ability to view communication just as a stream of bytes. This is nicer to deal with than random packet sizes, however, it rarely matches the communication model which will normally be a sequence of messages of user defined sizes.

Ports
: (Part of the transport layer in the TCP/IP model, but of the session layer in the OSI model) Ports are essentially ways to address multiple entities in the same location. For example, the first line of a postal address is a kind of port, and distinguishes between different occupants of the same house. Computer applications will each listen for information on their own ports, which is why you can use more than one network-based application at the same time.

Transport protocol comparison table

	UDP	**TCP**	**DCCP**	**SCTP**
Packet header size	8 Bytes	20 Bytes	Varies	12 Bytes + Variable Chunk Header
Transport layer packet entity	Datagram	Segment	Datagram	Datagram
Port numbering	Yes	Yes	Yes	Yes
Error detection	Optional	Yes	Yes	Yes
Reliability: Error recovery by automatic repeat request (ARQ)	No	Yes	No	Yes

Virtual circuits: Sequence numbering and reordering	No	Yes	Yes	Optional
Flow control	No	Yes	Yes	Yes
Congestion avoidance: Variable congestion window, slow start, time outs	No	Yes	Yes	Yes
Multiple streams	No	No	No	Yes

Examples

- AEP, AppleTalk Echo Protocol
- ATP, AppleTalk Transaction Protocol
- CUDP, Cyclic UDP
- DCCP, Datagram Congestion Control Protocol
- FCP, Fiber Channel Protocol
- FCIP, Fiber Channel over TCP/IP
- IL, IL Protocol
- iSCSI, Internet Small Computer System Interface
- NBP, Name Binding Protocol
- NBF, NetBIOS Frames Protocol
- SPX, Sequenced Packet Exchange
- RTMP, Routing Table Maintenance Protocol
- SCTP, Stream Control Transmission Protocol
- SCSI, Small Computer System Interface
- TCP, Transmission Control Protocol
- UDP, User Datagram Protocol

Categories: OSI protocols | Internet protocols | Transport layer protocols